AD-A111 203 AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOO--ETC F/6 15/7
A SIMULATION OF SECOND ECHELON AIR INTERDICTION.(U)
DEC 81 J E BENNETT
UNCLASSIFIED AFIT/GOR/OS/81D-3 NL

1 OF 1
AD
A111203

END
DATE
FILMED
03-82
DTIC

AD A111203

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFIT/GOR/OS/81D-3	2. GOVT ACCESSION NO. AD-A111 2	3. RECIPIENT'S CATALOG NUMBER 03
4. TITLE (and Subtitle) A SIMULATION OF SECOND ECHELON AIR INTERDICTION		5. TYPE OF REPORT & PERIOD COVERED MS Thesis
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) James E. Bennett 2LT		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Air Force Institute of Technology (AFIT-EN) Wright-Patterson AFB, Ohio 45433		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS		12. REPORT DATE December, 1981
		13. NUMBER OF PAGES 71
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

DTIC ELECTE FEB 2 3 1982 H

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

FREDRIC C. LYNCH, Major, USAF
Director of Public Affairs

Air Force Institute of Technology (ATC)

18. SUPPLEMENTARY NOTES

Approved for public release; IAW AFR 190-17 28 JAN 1982

Fredric C. Lynch

Frederic C. Lynch, Major, USAF
Director of Information

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

DCUBE
Second Echelon
Simulation
Anova

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

See Reverse

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

DTIC FILE COPY

A simulation model was developed to model second echelon air interdiction. The model was then validated using standard validation techniques. The results yielded by the simulation were statistically compared against the results of the DCUBE model, an analytic model employed by the Air Force to model the scenario. The statistical test yielded no significant difference between the results of the two models.

A SIMULATION OF SECOND ECHELON

AIR INTERDICTION

THESIS

AFIT/GOR/OS/81D-3 James E. Bennett
2LT USAF

AFIT/GOR/OS/81D-3

A SIMULATION OF SECOND ECHELON AIR INTERDICTION

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
in Partial Fulfillment of the
Requirements for the Degree of
Master of Science

by

James Evans Bennett, B.A.M.
2LT USAF
Graduate Operations Research

December 1981

Approved for public release; distribution unlimited

Preface

In this project, I have developed a model to simulate second echelon air interdiction. Although the final results of this project were not the expected outcome, this project served a useful purpose in teaching me many aspects of the job of an analyst and providing me an insight into the fundamental mission of the Air Force, to fly and flight.

I would now like to thank my thesis committee (Col Don Stevens (advisor), Lt Col Jim Bexfield and Major Jerry Armstrong), for without their guidance and patience this project would not have been possible. A special thanks goes to Major Jack Bogusch, a fellow student who took the time to educate me in the pilot's perspective of second echelon interdiction. Most importantly, though, I thank my wife, Cindy, and my one year old daughter, Allison, for the sacrifices they have made and for the inspiration only a family such as mine could provide.

James E. Bennett

Contents

List of Figures

Abstract

A simulation model was developed to model second echelon air interdiction. The model was then validated using standard validation techniques. The results yielded by the simulation were statistically compared against the results of the DCUBE model, an analytic model employed by the Air Force to model the scenario. The statistical test yielded no significant difference between the results of the two models.

A SIMULATION OF SECOND ECHELON AIR INTERDICTION

I Introduction

Background

In every military conflict involving the United States, reinforcements to the front line forces have played a key role in the outcome. In early history, the United States' ability to reinforce its own troops and sabotage the enemy through infiltration was paramount to the war efforts. In the twentieth century, however, the advent of military aircraft made direct attack on the enemy's supply lines and reinforcement channels (a.k.a. second echelon) possible. These attacks are known as second echelon air interdiction.

Second echelon air interdiction will be an important part of both adversaries' strategy in the event of a NATO-Warsaw Pact conflict in Europe. Due, however, to the large numerical advantage attributed to the Warsaw Pact ground forces, the ability of NATO to delay the arrival of enemy reinforcements could decide the battle. Therefore, efforts have been made to study the effects of second echelon interdiction and determine the "amount" of interdiction necessary to give a tactical advantage.

The DCUBE Model

Aeronautical Systems Division, (Mission Analysis Branch) at Wright-Patterson Air Force Base has developed an analytic

model to study second echelon air interdiction. This model was named DCUBE (D^3: delay, disrupt, and destroy) and consisted of two major parts, the air battle model and the ground battle model. In the DCUBE model, it is assumed that reinforcements will reach the FEBA (Forward Edge of the Battle Area) by traveling along "channels" from the rear echelon. Each of these channels is independent of the others. If, for any reason, traffic along one channel stops (i.e. to remove a damaged vehicle or repair a cut in the channel), there is no movement along that channel behind the forwardmost damage control site. All second echelon air attacks will take place against either vehicles in a channel or the channels themselves.

The DCUBE air battle model consists of a series of equations which calculate the total number of arrivals to the FEBA of AFV's (Armored Fighting Vehicles) during the duration of second echelon movement. This is accomplished by dividing the duration of second echelon movement into a number of distinct intervals, each of equal length. A calculation of the average arrival rate during the first interval is made. This is done using equations which take into account such elements as:

1. Initial spacing between AFV's,
2. Number of aircraft assigned to direct (AFV) and indirect (channel) attacks,
3. Probability that an aircraft survives to attack, and
4. Time it takes to repair the channels along which AFV's flow to FEBA.

A recursive relationship that gives the arrival rate for the subsequent intervals is then formed. This recursive relationship is based on the following principles.

1. Attacks are made at the beginning of each interval.

2. The number of AFV's in a channel at the end of an interval is equal to the number present in that channel at the beginning of the next interval. Vehicles which enter the channel are limited to the speed of the vehicles ahead of them.

3. Each direct attack sortie kills a constant fraction of the AFV's in the channel and each indirect attack makes a constant number of cuts in the channel.

4. No AFV's move in a channel while there is damage control to be completed.

The relationship is formed in a manner such that the output is the rate AFV's arrive at the FEBA during each interval. Therefore, since the length of each interval is known, the number of AFV's which reach the FEBA during each interval is also known. This output from the air battle model is the input to the ground battle model.

The ground battle model is a series of Lanchester equations which calculate the outcome of the force-on-force battle at the FEBA. These Lanchester equations are able to incorporate periodic reinforcements into the battle and, therefore, readily accept the output from the air battle model. The output from the ground battle model is the strength (in numbers) of the opposing ground forces during each period. From the total output from both models, therefore, one is able to ascertain the losses at the FEBA and determine who has the advantage in the conflict. Then, by varying different parameters in the

DCUBE model, one is able to see the effects different degrees of interdiction have on the ground battle.

Problem

While the DCUBE model has served a useful purpose to the Air Force in studying second echelon interdiction, there are problems which are inherent to some analytic models. From interval to interval, the percentage of aircraft surviving long enough to attack the second echelon remains the same in DCUBE. In a real world situation, the percentages of aircraft which are lost due to AAA will vary with time. This is due to the natural attrition of the anti-aircraft as the trucks are killed while on the way to the battle. In addition, the DCUBE model assumes that each aircraft surviving long enough to attack the second echelon 1) kills a constant percentage of available targets if assigned to direct attack and 2) creates a constant number of cuts in the channel if assigned to direct attack. Again, these assumptions would not hold in a real wartime scenario.

From these examples it is apparent that there are weaknesses in the DCUBE model. Non-constant percentages of kills and anti-aircraft attrition should be included in a model of second echelon air interdiction. This could be done through the use of simulation techniques. A simulation could be constructed that accounts for the DCUBE discrepancies. This simulation could also serve as a basis for further study in the area of air interdiction.

Another problem with the DCUBE model is the use of the exponential approximation to the binomial distribution. Although this approximation was useful before the advent of digital computers, with the present hardware available, the extra time to compute the exact probability is no longer a problem. This approximation should be removed from the model due to potential problems with certain values which could be input into DCUBE.

Objectives of Thesis

To help further the understanding of second echelon interdiction and provide insight into the problem with DCUBE, the topic of this thesis will be a simulation of the DCUBE air battle model. The objectives of this thesis are as follows:

1. The DCUBE air battle model will be simulated. The output from the simulation air battle will be documented in the program identically to that of the DCUBE model. Both models will be run and for similar cases (i.e. all clear weather in both models), the results of the two models will be statistically compared to see if there is a difference. If differences exist, the reasons for the difference will be brought out and explained.

2. The model will be verified using standard verification techniques. Parameters will be varied and, using the same random numbers, a thorough check will be made to see if the model acts as intended.

3. The simulation model will be documented in a manner such that it will provide an effective means for studying effects of new weapons and strategies.

4. Recommendations for further work in this area will be made. Potential research topics will be presented.

Summary

This text will present an explanation of the equations which make up the DCUBE model. This will be done in a manner such that the DCUBE will be presented to the reader along with the identification of the model's potential weaknesses. Once the DCUBE has been summarized, the model simulating the same scenario will be presented. Included in this will be the assumptions made for the simulation along with an explanation of the code involved.

After both models have been explained, a statistical analysis of the results of the two models will be presented. This analysis will include a summary of the simulation output and a comparison of the results of the two models when both are run using the same inputs. This should provide a basis for a conclusion on the relative worth of the two models.

II The Analytic Model of Second Echelon Air Interdiction

In any conflict, the arrival of reinforcements to the battle plays a key role. The ability of one side to delay, destroy or disrupt movements in the enemy second echelon can swing the course of the battle. Because of this interest in second echelon movements, models have been constructed to study the effect of second echelon disruption.

Of particular interest in this thesis is the DCUBE (D^3: delay, destroy and disrupt) model employed by ASD Mission Analysis. The following text will attempt to explain the analytics involved in the model. In the process, points which appear t be weaknesses in the model will be brought out. This should help to further the understanding of the model.

The DCUBE Model

The DCUBE model considers both the immediate and overall effects of second echelon interdiction. The immediate effect of second echelon interdiction is the damage to the enemy reinforcement effort. This damage can be either vehicle of roadway destruction. The extent of the destruction is determined by weapon size, weapon accuracy and number of attacks.

Once the immediate effects have been calculated, they are translated into overall effects. This is accomplished by determining the rate at which vehicles arrive at the FEBA, and subsequently calculating the effect of the arrivals on the

battle. The measure of merit of the DCUBE model, therefore, is the outcome of the ground battle.

The DCUBE model employs two different sub-models during a run. The first, the arrival rate model, calculates the immediate effects of interdiction and determines the rate of arrivals to the FEBA. Then the second model, the ground battle model, uses the arrival rate as an input and calculates the outcome of the battle. One is then able to study the effects new weapons or strategies may have on interdiction efforts.

The subject of this chapter is the arrival rate model. The analytics involved should serve as a basis for further study.

The Arrival Rate Model

The arrival rate model uses the following assumptions:

1. the flow of reinforcements to the FEBA is accomplished along Q channels,
2. reinforcements are distributed evenly among the Q channels, and
3. attacks are made on the channel along which they will be most effective.

The formula upon which the model is based is

$$Z=Q \int_0^D R(t)\, dt \tag{1}$$

where the following variables are use.

Z the total number of arrivals to the FEBA

D the total length of time over which the interdiction takes place

R(t) the function which represents the instantaneous arrival rate at time t, $0 < t < D$

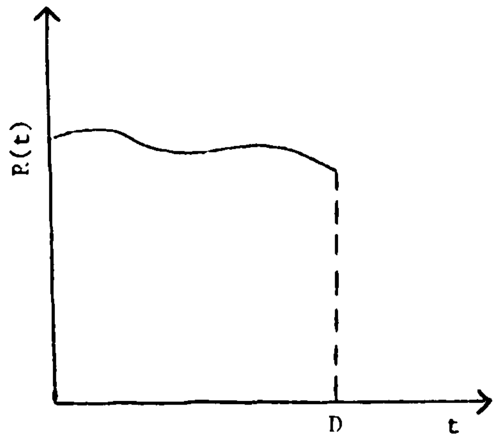

Figure 1 Graphical Representation of Arrival Rate

Equation 1 gives the shaded area under the curve (FIGURE 1) which represents total arrivals to the FEBA. The difficulty, though, arises in coming up with an expression for R(t). Approximation of a continuous curve such as R(t) cannot be made due to unpredictable changes, such as the number of vehicles in the channel. Therefore, DCUBE employs the following summation to calculate Z:

$$Z = Q \sum_{j=1}^{(D/dt)} R_j \cdot dt \qquad (2)$$

In this equation, D is divided into intervals, each of equal length dt. R_j is the average arrival rate during interval j and (D/dt) represents the least integer greater than c equal to D/dt. Therefore, it remains only to calculate R_j for each j. To do this, a recursive relationship will be formed from the following sequence of events.

1. The number of vehicles in a channel at the start of the $j^{\underline{th}}$ interval is equal to the number present after the air attack during interval $j-1$, minus the number which reached the FEBA, and plus the number which entered the channel following the air attack. The net gain of targets (positive or negative) in the process is X_j.

2. Aircraft attack either vehicles in the channels (direct attacks) or the channels themselves (indirect attacks). The model assumes that a constant percentage of the targets available are killed by each direct attack and a constant number of cuts are made by indirect attacks. (I feel that these assumptions take away from the model. Given the human and mechanical problems which may arise, the variability in the number of successful attacks from interval to interval would be too great for it to be approximated by a constant.) These assumptions necessitate the assumption of independence between sorties.

3. There is no movement in a particular channel behind a cut in that channel.

4. Some of the targets will arrive at the FEBA during interval j at rate R_j.

Letting

V_j be the average velocity of vehicles along a channel during interval j, and

A_j be the average spacing between vehicles during interval j.

R_j can be formed using

$$R_j = \frac{V_j}{A_j} \tag{3}$$

It is assumed that all vehicles in a channel during an interval are equally spaced. Because of this assumption, the average spacing during interval j can be derived. Letting N_j be the number of vehicles surviving in a channel after the

air attack during interval j, it is apparent that

$$A_j = \frac{A_o \cdot N_o}{N_j} \tag{4}$$

For example, suppose there were originally 200 vehicles in a channel, each 200 meters apart. Suppose there are now 400 vehicles in the channel. They are (200·200)/400=100 meters apart.

Since any cut in a channel stops the flow along that channel behind the cut, vehicles in the channel will move at either V_o or stand still. Therefore, the expected velocity during interval j is given by

$$V_j = 0 \cdot P_{Dj} + V_o(1-P_{Dj}) = V_o(1-P_{Dj}) \tag{5}$$

where

P_{Dj} is the probability a vehicle experiences a delay during interval j. The time a target spends in a channel in the absence of air attack (T_o) is a factor of certain inputs.

$$T_o = \frac{A_o \cdot N_o}{V_o} \tag{6}$$

For example, if a channel is 40,000 meters long, and there are 200 vehicles in the channel, each moving at 4000 m/hr, vehicles spend (200·200)/4000=10 hours in the channel.

Using equations 3, 4, 5 and 6 yields:

$$A_o = \frac{T_o \cdot V_o}{N_o} \Rightarrow R_j = \frac{V_j}{A_j} = \frac{V_o(1-P_{Dj})N_j}{A_o N_o} = \frac{V_o(1-P_{Dj})N_j}{\frac{T_o V_o N_o}{N_o}} =$$

$$\frac{(1-P_{Dj})N_j}{T_o} \tag{7}$$

To fully develop R_j as a function of inputs, P_{Dj} and N_j must yet be determined.

Since N_j is the number of vehicles surviving the air attack during interval j,

$$N_j = (N_{j-1}+X_j)\cdot(1-P_{Kj}) \tag{8}$$

where

X_j is the net gain in targets between interval j-1 and the start of interval j, and

P_{Kj} is the probability that a given target is killed during interval j.

That is, the number of possible targets in the $j^{\underline{th}}$ interval is the sum of the targets remaining after the $j\text{-}1^{\underline{st}}$ attack (N_{j-1}) plus the net gain in targets (X_j). To get N_j, a constant fraction ($1-P_{Kj}$) of the sum is taken. Therefore, we further need to isolate N_j and P_{Kj}.

The net gain in vehicles present in the channel from j-1 to j is simply the number of vehicles present in the channel during j-1 (after the j-1 air attack) plus the number which entered the channel after the air attack, minus the number

which managed to reach the FEBA. Since vehicles enter the channel during j-1 according to the average speed during j-1, divided by the spacing, then

$$X_j = \frac{V_{j-1}}{A_o} \cdot dt - R_{j-1}dt \tag{9}$$

A_o is used as the spacing for entering vehicles because it is assumed that vehicles will only enter the channel while movement is allowed. Therefore, since vehicles move at V_o when unhindered, they will be spaced according to the original length.

From this, it can be concluded that

$$R_j = \frac{V_j}{A_j} => V_{j-1} = R_{j-1} \cdot A_{j-1} => X_j = \frac{R_{j-1} \cdot A_{j-1} \cdot dt}{A_o} - R_{j-1} \cdot dt$$

$$=> X_j = (\frac{A_{j-1}}{A_o} - 1) \cdot R_{j-1} \cdot dt \tag{10}$$

Combining equations 10 and 4 yields

$$X_j = (\frac{N_o}{N_{j-1}} -1) \cdot R_{j-1} \cdot dt \tag{11}$$

At this point, only P_{Kj} is needed to conclude the calculation of N_j. Letting P_{KT} be the probability that a target, T, is killed during an interval, it can be seen that

$$P_{KT} = \frac{K}{N_o} \tag{12}$$

where

N_o	is the number of targets initially in the channel and
K	is the number of targets killed out of a group of size N_o.

(K and N_o are both inputs). Now, assuming there are S sorties attacking a channel during j, then

$$. \quad P_{Kj} = 1-(1-P_{KT})^S \tag{13}$$

At this point, the authors of the DCUBE model approximate equation 13 by

$$1-(1-P_{KT})^S \approx 1- \exp(-S\cdot P_{KT}) = 1- \exp(-S\cdot \frac{K}{N_o}) \tag{14}$$

Using

B_{10}	as the number of aircraft initially assigned to direct attack,
U	as the number of sorties per plane per hour, and
E_j	as the probability that a given aircraft survives to attack during interval j,

yields

$$S = \frac{B_{10}}{Q} \cdot U \cdot E_j \cdot dt, \tag{15}$$

(Suppose there are 2000 planes assigned to direct attacks (B_{10} = 200) and there are 5 channels (Q=5). Let each aircraft fly 2 sorties/hour, and have a probability of .7 of attacking. Assuming each interval is 1 hour long, the number of attacking sorties (15) is given by $\frac{200}{5} \cdot 2 \cdot .7 \cdot 1 = 56$.)

with

P_A as the probability that a given sortie fails to survive long enough to attack its target, then

$$E_j = (1- P_A)^{U \cdot j \cdot dt} \approx (\exp(-U \cdot P_A \cdot dt))^j. \tag{16}$$

Combining 13-16 yields

$$P_{Kj} \approx 1- \exp(\frac{-K \cdot B_{10} \cdot U \cdot E_j \cdot dt}{N_o \cdot Q}) \tag{17}$$

Combining 8, 11 and 17 then yields a recursive form for N_j,

$$N_j = N_{j-1} + (\frac{N_o}{N_{j-1}} - 1) \cdot (R_{j-1} \cdot dt) \cdot (\exp(-K \cdot B_{10} \cdot U \cdot E_j \cdot \frac{dt}{N_o \cdot Q}) \tag{18}$$

Now, only an expression for P_{Dj} is needed to form a recursive expression for R_j(eq.7).

Sovling for P_{Dj}

Let

dT_j be the expected time to clear the channel as of the end of the air attack in interval j (hrs.), and

dT_{1j} be the expected time to perform damage control upon all killed vehicles as of the end of the air attack in j, and

dT_{2j} be the expected time required to repair cuts on the channel as of the end of the air attack in interval j (hrs.).

The probability that there is a delay during the $j^{\underline{th}}$ interval can be expressed as

$$\frac{dT_j}{dt} \approx 1- \exp(-\frac{dT_j}{dt}) \tag{19}$$

Remembering that dT_{1j} and dT_{2j} are delays associated with direct and indirect attacks, the DCUBE model assumes a common work force to clear a channel, so

$$dT_j = dT_{1j} + dT_{2j} \tag{20}$$

(i.e. the work force will first clear away dead vehicles and then repair the channels).

Using

C_D	as the length of time it takes to clear a dead vehicle (per remaining vehicle) and
Y_j	as the number of killed targets that have not yet been cleared (i.e. subject to damage control) immediately following the air attack in interval j,

dT_{1j} can be expressed as

$$dT_{1j} = C_D = \frac{Y_j}{N_j} . \tag{21}$$

Suppose it takes $\frac{1}{2}$ hour for the crew of 1 remaining vehicle to clear away a dead vehicle ($C_D=\frac{1}{2}$). Also suppose there are 10 dead vehicles ($Y_j=10$) and 50 live vehicles ($N_j=50$) at the end of the air attack during interval j. Then the delay associated with dead vehicles (dT_{1j}) during interval j is $\frac{1}{2}$ hr * 10/50=1/10 hr.

Y_j can be comprised of either kills during the $j^{\underline{th}}$ air attack or vehicles which were killed during j-1 and are still blocking the road. Therefore, with P_{Uj} as the probability that a given dead vehicle has not yet been cleared during time interval j,

$$Y_j = P_{Kj} \cdot (N_{j-1} + X_j) + P_{Uj} \cdot Y_{j-1} \tag{22}$$

Here, $P_{Kj} \cdot (N_{j-1} + X_j)$ represents the "new" dead vehicles which are blocking the channel and $P_{Uj} \cdot Y_{j-1}$ represent kills from earlier intervals which are still blocking the channel. The authors present the formula

$$P_{Uj} = \frac{dT_{1j-1} - (1- \exp(dT_{1j-1}/dt)) \cdot dt}{dT_{1j-1}} \tag{23}$$

for the probability a given dead vehicle from interval j-1 has not been cleared following the air attack during interval j. The rationale behind equation 23 is as follows. The expected delay associated with direct attacks during interval j-1 is dT_1. $(1-\exp(dT_{1j-1}/dt)) \cdot dt$ gives the time during j-1 that will be needed to clear the channel. Therefore, $dT_{1j-1} - (1-\exp(dT_{1j-1}/dt)) \cdot dt$ gives the length of the delay associated with direct attacks during j-1 that must be carried over to j.

Let

C be the number of distinct cuts per channel per sortie, and

W be the length of time a channel is cut per cut.

Using the same argument as for dT_{1j}, it can be shown that

$$dT_{2j} = C \cdot W \cdot B_{20} \cdot U \cdot E_j \cdot \frac{dT}{Q} + (dT_{2j-1} - (1-\exp(-dT_{2j-1}/dt))dt) \tag{24}$$

represents the length of the delay associated with indirect attacks during j-1 that must be carried over to j.

Remembering that

$$R_j = (1-P_{Dj}) \cdot \frac{N_j}{T_o}$$

all the expressions needed to form the recursive relationship have been developed.

A summary of the definitions presented in this chapter and a mathematical presentation of the DCUBE arrival rate model appear in Appendices A and B respectively.

III The Simulation of Second Echelon Air Interdiction

The simulation of the disruption of second echelon movement considers the same scenario as the DCUBE model. Trucks move along channels leading from the rear echelon to the FEBA. These trucks move according to a spacing between vehicles and a speed which are dictated by the current weather conditions. As the trucks move along the channels, they are subject to attack from enemy aircraft.

The simulation model considers three seperate channels along which vehicles may move to the FEBA. These channels are assumed to be sufficiently far apart to avoid a single plane having the opportunity to choose from more than one channel to attack. The channels extend from 45000 meters behind the FEBA up to a point 15000 meters behind the FEBA. The reason the channels "end" at 15000 meters is that 15 kilometers is the maximum range of present artillery and the simulation considers second echelon attack from the air only.

When an attack takes place and vehicles moving toward the FEBA are destroyed, there is a delay associated with each kill for vehicles behind damage points. The reason for the delay is the nature of the attack. Airplanes will be coming in at a high enough speed so that little time is available for vehicles to leave the roadway for cover. Therefore, if any of the vehicles are destroyed in the attack, the destroyed vehicle(s) will be on the roadway and must be cleared before movement can resume. The time it takes to clear away the damage is a func-

tion of the number of surviving vehicles in the immediate area. Once the damage is cleared, remaining vehicles move on to the FEBA.

The simulation is an event oriented program which centers around the next event associated with each airplane. The events which a plane can encounter are as follows: planes may

1. be attacked by surface-to-air missiles over the FEBA on the way toward the second echelon,
2. search for targets over the second echelon,
3. attack targets on the roadway,
4. encounter surfae-to-air missiles over the FEBA on the way back from the second echelon, or
5. return home for weapons and fuel.

Depending on the time and type of the next event associated with each airplane, a routine simulating one of the preceeding events will be called, the event simulated, and a new next event time and next event will be assigned. After each event takes place; the movement of vehicles along the channel is calculated. This process continues until all vehicles have either reached the FEBA or been destroyed.

Format

For the purpose of the simulation, an imaginary grid was placed over the second echelon. The grid was 3x30 with each cell having a width of 1000 meters and a height of 5000 meters. Each row contained one and only one of the channels leading from the rear echelon. The assignment of numbers to the grid was from the perspective of the trucks moving to the FEBA. (i.e. When a truck initially enters the second echelon, it is in column 1. After it has moved 1000 meters closer to the FEBA,

it is in column 2) The northernmost grid row was numbered 1, the middle row was numbered 2 and the southernmost row was numbered 3. Therefore, if a truck was in row 1 and column 5, it is in the northernmost channel and between 25,000 and 26,000 meters from the end of the channel.

Variables

The variables to be used in the simulation are:

ANGLE	-	the angle of attack taken by an attacking plane against vehicles in a channel
AO	-	the spacing between vehicles in the channels (for calculating movement) at a given time
AODA	-	the normal daytime spacing between vehicles in the channels
AONI	-	the normal nighttime and bad weather spacing between vehicles in the channels
ARSPED	-	the cruising airspeed of the airplanes used in the second echelon attack
ATTACK	-	the point of attack (i.e. optimal attack point) of planes attacking vehicles in a channel
CEP	-	the maximum assumed error associated with an attack (i.e. assumed distance away from attack point which weapon will hit)
CLDAM(H,J)	-	the time damage point "J" on channel "H" will be repaired
CLVEH	-	vehicle minutes required to clear away a damaged vehicle
CNTROD	-	the actual point of impact on a channel of an attacking plane's weapons
DAM(H,J)	-	a status variable indicating whether or not there is damage in row H, column J
DAMPT(H,J)	-	the exact location of the $J^{\underline{th}}$ damage point on row H

DAYBRK	-	the length of time from the total darkness of night to full daylight
DAYTIM	-	the length of time of full daylight
DIST	-	the effective distance of an attacking plane's cluster bomb units from their point of impact
FLYTIM	-	the length of time it takes for a plane to fly from its home base to the FEBA
KILLTTL	-	the total number of trucks killed in all attacks during the duration of second echelon movement
KLPL	-	the total number of planes killed in all attacks during the duration of second echelon movement
LMN	-	the total number of trucks which have reached the FEBA
LSTEVT	-	the time at which the preceeding event took place
MMM	-	the sum of the number of vehicles killed in attacks and the number which have reached the FEBA
NPLATT	-	the number of times any place came under attack during the second echelon movement
NTATT	-	the sum of the numbers of trucks in a channel at the time of attack in that channel
NTFALL	-	the length of time from full daylight to complete darkness of night
NTRUC(H,J)	-	the number of trucks in column J of row H
NTRUCK	-	the total number of trucks to be moved through the second echelon
NUMDAM(H)	-	the number of damage points on row H at a particular time
NUMPLA	-	the total number of planes available to attack the second echelon
NWEP(J)	-	status variable indicating whether or not plane J has weapons

NXEV(J) - this attribute of plane J indicates the type of the next event associated with plane J (The following chart indicates the values which NXEV(J) may assume and their meanings.)

NXEV(J)	INTERPRETATION
1	PLANE(J)'s next event will be to return home for refueling and weapons
2	PLANE(J)'s next event will be to search for targets over the second echelon
3	PANE(J)'s next event will be to encounter SAM's over the FEBA
4	PANE(J)'s next event will be to attack vehicles in the second echelon
5,6,7,8	these values indicate a change in the weather conditions of the theatre

NXEVT(J) - the time of the next event associated with plane J

NXTEVT - the time of the next event to take place

PKAA(K) - the probability that the $K^{\underline{th}}$ plane involved in an attack is shot down by AAA fire

PKAAA - the probability that a single AAA battery can shoot down an airplane at a given time

PKAADA - the probability that a single AAA battery can shoot down an airplane during the day

PKAANI - the probability that a single AAA battery can shoot down an airplane during the night

PKSAM - the probability that a plane is killed by a SAM attack over the FEBA

PKTRUC - the probability a truck within the effective range of a plane's cluster bomb unit's point of impact is killed

PLNCOL(J) - the number of the column over which plane J is flying

PLNROW(J) - the number of the row over which plane J is flying

TRAAA(J) - the status variable indicating whether or not truck J is pulling AAA

TRUCK(J) - the location of truck J, given in meters away from the entrance of its channel

TRUCR(J) - the number of the channel in which truck J is located

VO - the speed which trucks in the channels are moving at a particular time

VODA - the speed which trucks in the channels move during daylight

VONI - the speed which trucks in the channels move during the night

The Simulation Model Code

The simulation model is set up in a series of routines, each simulating a unique portion of the second echelon interdiction scenario. These six routines are:

1. Initial Aircraft Takeoff
2. SAM Attacks over the FEBA
3. Search for Targets by Aircraft
4. Attack of Targets by Aircraft
5. Aircraft Turnaround
6. Truck Movement Toward the FEBA

The purpose of this section is to explain the FORTRAN code used to simulate each of these events. This explanation will include a discussion of the interaction between events and the assumptions made concerning the events.

Initial Aircraft Takeoff

When the conflict begins, all available aircraft with the mission of interdicting the second echelon will take off and head toward the FEBA. Planes will fly in groups of two, a

commander and his wingman, and both airplanes in a flight will take off simultaneously. It is assumed that all aircraft used for this mission will take off from the same base, approximately 170 miles behind the FEBA and it will take one minute for a flight of two airplanes to take off given another flight has just taken off. Following takeoff, the next event associated with each airplane is to be attacked by surface-to-air missiles while flying over the FEBA.

DO loop 10 simulates this by setting the values of the attributes NXEVT and NXEV for each plane. Since each plane will next encounter enemy SAM over the FEBA, NXEV is set to 3 for each plane. In determining the time of the next event for each plane, two elements must be taken into account; when they take off and the time it takes to reach the FEBA. Letting Z represent the flight number of each group of two airplanes allows NXEVT to take on the value

NXEVT(J)=Z + FLYTIM + X/1000.

(For example, the first flight (Z=1) will reach the FEBA in 1 + FLYTIM minutes. Adding X/1000 insures that later in the simulation, no two flights will have the same value in their NXEVT attributes.) The DO loop, therefore, initially groups planes 1 and 2, 3 and 4, 5 and 6,...., NUMPLA-1 and NUMPLA, into flights.

Determining The Next Event

This routine determines which planes are involved in the

next event. L and K are the two variables which will contain the numbers of the two planes which will take part in the next event and, therefore, must be initially set to zero. L will remain at zero if only one plane is involved in the next event. Setting LSTEVT to the value currently in NXTEVT gives a storage place for the time of the preceeding event. DO loop 20 then checks the value of the attribute NXEVT for each plane and places the smallest value into the variable NXTEVT. The number(s) of the plane(s) involved are then placed in K (and L). Then, by checking the value of attribute NXEV(K), the appropriate routine to simulate the next event is called.

SAM Attack Over The FEBA

If the value in NXEV(K) is 3, the next event the model will simulate is a SAM attack over the FEBA. If NWEP(K) is equal to zero, this implies that plane(s) K (and L if K's wingman has survived up to this point) have finished their mission over the second echelon and are returning home. If NWEP(K) is equal to one, K and L are heading toward the second echelon.

If the planes are headed toward the second echelon, it is known that two planes are involved in this event because planes will only leave their home base in flights of two. Therefore, two random numbers are drawn and compared to PKSAM, the probability an aircraft is killed by SAM fire. If both airplanes survive, they continue on toward the second echelon

and their attribute NXEV assumes the value 2 (search for targets) and their next time is determined by NXEVT(K) + 15000/ARSPED. This is due to the fact that the area to be searched begins at 15000 meters behind the FEBA and the planes can fly 15000 meters in 15000/ARSPED minutes. (For example, suppose the time is presently 100 and ARSPED is 1000 meters/ minute. Then the time of the next event for plane K is 100 + 15000/10000 = 101.5) If one or more of the planes is killed, then the next event for the survivor, if any, is 1 (aircraft turn around) and the next event time is given by NXEVT(K) + FLYTIM. To insure that an airplane killed in the attack will not be considered later in the simulation, its attribute NXEVT is given a large value, 1000001.

If NWEP(K) is equal to zero, this implies the number of planes involved may be one or two. Therefore, depending on the number of planes in the event, one or two random numbers are drawn and compared against PKSAM. Any airplanes killed are given the value 1000001 in their attribute NXEVT and will no longer be considered in the simulation. If any planes survive this attack, their next event will be 1 (return home) and their next event time will be NXEVT(J) + FLYTIM.

Aircraft Turnaround

If the value in NXEV(K) is 1, the next event the model will simulate is aircraft turnaround. There are two situations which returning aircraft may encounter at the home base. If two planes, K and L, are returning simultaneously, these

two airplanes will leave together as a flight as soon as they are refueled. The average time it takes for planes to be refueled and rearmed is assumed to be 25 minutes. Therefore, since the next event for planes when they arrive home will be to encounter SAM over the FEBA(3), the time of the next event for the planes in NXEVT(K) + 25 + FLYTIM.

However, if only one of the two planes (plane K) from a flight return, then plane K will team with a single plane, if available, which is waiting for a partner. If there is a plane waiting, then these two planes will leave together as a flight as soon as the last plane is refueled and rearmed. Therefore, their next event time will be NXEVT(K) + 25 + FLYTIM. If there is no plane waiting, then the arriving single plane will have to wait at the base until another single plane arrives.

Search Routine

If the value in NXEV(K) is 2, the next event the model will simulate is planes K and L's search for targets while flying over the second echelon. The area to be searched by the aircraft is given by their position in the grid pattern over the second echelon. The aircraft will search the particular area over which they are flying and if there are no available targets or if there is already damage in the area, the planes will continue its search.

The model handles this situation by checking the area on the grid defined be PLNROW(K) and PLNCOL(K), the row and

column location of plane K. If, however, the preceeding event for planes K and L was the SAM attack over the FEBA, the row over which they will fly will be undefined. Therefore, this routine must first assign rows to those planes which are just entering the second echelon.

If there is no damage anywhere in the second echelon or if there is damage in all three rows, there is no choice of rows which will provide a better target area for the planes than any other row. Therefore, a random choice among the rows which the planes may fly is used. If there is damage on only one of the rows, a random choice among the other two is made. However, if there is only one row without damage, that is the row chosen for planes K and L to fly down. In addition, since the planes are just entering the second echelon, the column they will be over will be 30, so PLNCOL(K) is set to 30. (Recall that the column number is from the truck's perspective and that each column is 1000 meters wide. Therefore, when planes enter the second echelon they will be over the last of the thirty columns from the trucks perspective.)

Once the location of the planes is established, the routine checks the area defined by the row and column numbers of the planes. If there are three or more trucks in the "PLNROW(K), PLNCOL(K)" area and no uncleared damage, the next event for plane K and L will be 4 (attack) and the routine simulating the attack will be immediately called. (In general, if there are three or more vehicles in a given area, the pilots of the searching aircraft are guaranteed detection either visually

or through the use of infrared scopes.) The reason the attack routine is immediately called is because the time between identification of targets and the actual attack is very small and for simulation purposes assumed to be zero.

If there are no targets available, there are two situations which may arise. If the searching planes have reached the end of the second echelon (i.e. PLNCOL(K) is equal to one), then due to the fuel limitation the planes must return home. Therefore, their next event time will be 3 (encounter SAM over the FEBA) and since the planes are 45000 meters behind the FEBA, their next event time will be NXEVT(K) + 45000/ARSPED. As an indicator to the SAM routine that the aircraft are returning home, the values of NWEP are set to zero for both planes. However, if the planes are in a position to continue their search, their next event will be to continue searching (2) and they will search the next column (PLNCOL(K) - 1). The time of the next event will be NXEVT(K) + 1000/ARSPED because the columns are 1000 meters wide.

Weather Effects

This routine updates the status of the weather, including day versus night, in which the model operates. The weather changing forces changes in AO, VO and PKAAA. The changes in AO and VO will take place immediately any time the status of the weather changes. This is because the total change must be in effect before the new weather sets in completely and the formation of the column is lost. If, however, the change

to be accomplished is from day to night, PKAAA will move from PKAADA to PKAANI gradually. This will be the case in the modeled scenario because the visibility will reduce gradually and AAA will lose some of its effectiveness slowly.

The model handles the changes according to the values in NXEVT and NXEV for some J. The times of all the changes must be input by the user of the model. Since the values in NXEV correspond to the next event for an airplane for all $J \leq NUMPLA$, the indices for the arrays must be greater than NUMPLA. The exact format will be shown in the inputs section of the next chapter.

Second Echelon Attack

If planes searching the second echelon acquire targets, then the next event the model will simulate is the attack on the second echelon. This attack will be made using cluster bomb units with an effective area of 200 by 400 meters. This rectangle will "lie" over the target area with the longer axis parallel to the flight path taken by the firing aircraft during its attack run. All trucks lying in the rectangle are subject to the weapon and have a chance to be killed. The location of the centroid of the cluster bomb unit's effective area will be determined by the pilot's choice of the optimal point of attack and the CEP (as defined).

To simulate this, the model must first determine the forwardmost point which the command pilot could possibly choose to attack. To do this, the location of the forwardmost truck

(FRWARD) in the group of trucks which the pilot spotted is found. (For example, suppose there was damage in column 28 at the point 27750 meters into the channel and no damage behind that point. In this case, the pilots would have passed on the attack on column 28 due to the damage and gone on to search column 27. Then, after acquiring targets in 27, the attack routine would have been called. However, due to the fact that the trucks will try to maintain a constant spacing, there will be many undamaged trucks lined up behind the 27750 meter point. Therefore, the forwardmost point at which an attack would be made would be 27750.) In an attempt to maximize the truck kills, the pilots would choose a point approximately 500 meters behind forwardmost point to attack so as not to waste their weapons by covering an area already hit.

Once the attack point is chosen, the actual attack on the channel must be simulated. (Since each airplane's attack is independent, the two attack runs and their effects will be considered separately.) The first items to be considered in an attack are the angle of attack taken by the attacking airplane, and the error from the optimal point of attack. To determine the error associated with the weapons drop, two random numbers are drawn. The first random number indicates whether the weapon hit it the right or left of the target and the second, by multiplying it by CEP, determines the actual point of impact. Once the actual point of impact is calculated, the angle of attack must be calculated to give a means of determining the numbers of trucks subject to the weapons. (To illustrate this

point, consider a rectangle with a straight line drawn through its centroid. As the rectangle is rotated around its centroid, the amount of the line "covered" by the rectangle will vary. In the context, of simulation, the calculation of the angle of attack will yield which of the trucks in the "straight line" will fall in the rectangular area of coverage associated with each plane's attack.) After the exact area of coverage is found, a random number is drawn for each truck within that area and compared against PKTRUC. If the random number is greater than or equal to PKTRUC, the associated vehicle is unhurt and there is no effect on second echelon movement. If the random number is less, however, that truck is destroyed and in its exact location is a damage point which must be subjected to damage control. The damage point is then placed in its proper location in the DAMPT matrix in computer storage. Its proper place is determined by the row it is on (H) and its location compared to the other damage points in that row. For example, if the new damage point is closer to the FEBA than any other, its location is put in the DAMPT(H,1) position and all the other damage points are in their new positions. In addition, all the damage points clearance times are put in the CLDAM array in the location corresponing to their damage point position in the DAMPT array.

After the attack is simulated for both planes and all the new damage points are accounted for, the attack routine calculates the amount of time it will take to clear away the new damage so movement behind the damage can continue. This is

done by first counting the number of surviving trucks within 1000 meters behind the forwardmost damage point. The crews on these vehicles are responsible for the damage control. Therefore, the amount of time until the damage points are cleared is given by the number of newly killed vehicles multiplied by the amount of time it takes to clear away a damaged vehicle per survivor (CLVEH) divided by the number of survivors. (For example, suppose there are six damage points, ten survivors and it would take one survivor 5 minutes to clear away one dead vehicle. Then the total amount of time it would take to clear away the damage and let total second echelon movement resume would be 6·5/10=3 minutes.) Once the amount of time it will take to clear away all the damage is computed, the time that the "simulation clock" will recognize the damage as being cleared is recorded in the CLDAM array in the location corresponding to the new damage point's location in the DAMPT array. (i.e. If the time of the attack is 35 and it will take 3 minutes to clear away the damage, at time 38 all damage control will be completed. Hence, if there are two new damage points and they are in 1, 3 and 1, 4 positions of DAMPT, then the values in CLDAM (1,3) and CLDAM(1,4) will be 38.)

Upon completion of the attack and damage calculations, this routine simulates the subjection of the attacking aircraft to AAA fire from surviving vehicles in the channel. In the present Warsaw Pact arsenal, there are many different AAA weapons which can be pulled behind vehicles during troop and support movement. This AAA can be fired while the trucks are

moving and uses conventional 20mm-50mm cannons. The model identifies which trucks are pulling AAA by putting the value 1 in the truck attribute TRAAA(J) for the $J^{\underline{th}}$ truck if it has AAA capability. The model assumes that only AAA within 500 meters of the attack point (which will also be approximately the point where the aircraft are closest to the ground) are able to fire on the planes and will only have time to fire one burst. This is due to the limit which AAA has in firing at targets low in the horizon and the fact that when AAA further away will be able to fire, the aircraft will have hit afterburners and AAA will no longer be effective. The model simulates this by first determining the exact number of vehicles pulling AAA within 500 meters of the attack point. It is assumed that 50 percent of the AAA able to fire will fire at each aircraft. Therefore, the probability each airplane is killed, if there are no AAA batteries available, is $1-(1-PKAAA)^{n/2}$ A random number is then generated for each airplane and compared to the probability the plane is killed. If an airplane is killed, it is given a very high value in its attribute NXEVT so it will not be considered later in the simulation. If it is unaffected by AAA fire, its next event time is given by NXEVT(K) = (15000+(30-PLNCOL(K))* 1000) and its next event will be to encounter SAM over the FEBA on the way back to the home base.

Truck Movement

After each event for a truck has been completed, except

for the search routine during which a target is acquired, the simulating truck movement is called. Using the time between the preceeding two events, there is a distance which trucks may have moved during that time if there is no damage inhibiting movement. This routine calculates the movement made by each truck and also determines different status variables for the complete simulation.

At this point in the simulation, the model checks the values in the CLDAM array to check on whether or not any damage control has been completed since the preceeding event. If it has, the associated damage points and repair times are removed from the DAMPT and CLDAM arrays and the arrays manipulated to reflect their removal. (Suppose there are damage points only one one row (row z) and there are three. Therefore, the (z,1), (z,2), and (z,3) positions are the only cells in the arrays which contain non-zero values. If the damage control for DAMPT (z,1) has now been completed, then the value in (z,1) is set to zero. Then the value in (z,2) is moved to (z,1), (z,3) is moved to (z,2) and (z,3) is set to zero. This is done for both the DAMPT and CLDAM arrays.)

The model then simulates the truck movement along the channels. To do this, the amount of time between the preceeding two events is calculated and multiplied by the velocity which trucks are presently moving. This yields the maximum distance trucks may move if unhindered (TMOVE). Movement is only allowed between damage points and each truck is considered separately. Considered first is the area between the end of the channel and

the first damage point on row 1. Since there is no limit on how far that truck may move, in the channel, its TRUCK attribute assumes the value given by MIN (FRONT, TRUCK(J) + TMOVE) where FRONT is 30000. If this truck leaves the channel (i.e. passes the 30,000 meter point), then FRONT remains 30,000 and movement for the next truck is calculated. If at any time a truck, say truck 100, remains in the channel after movement, no truck behind it may move any farther than TRUCK(100)-AO. Hence, FRONT becomes TRUCK(100)-AO. This process continues for all the trucks ahead of the first damage point on row 1. For trucks behind the first damage point only movement to maintain the spacing between vehicles is allowed and the routine moves the trucks according to the AO presently in use.

Once this process is completed for all three channels, the routine calculates the new values for all cells in the arrays NTRUC, DAM and NUMDAM. Then, unless all trucks have either left the second echelon or been killed, the model returns to the routine which determines the next event and the simulation continues.

Conclusion

After the code to simulate the events presented in this chapter was developed, verification was required. The next chapter presents the results of the verification process.

IV Verification of the Simulation Model

Inputs

Initially, all variables in the model must be set to zero, except for those which are inputs into the simulation. Those variables for which values must be input by users of the program (and the values used in validation of the program) are: .

INPUT VARIABLE	CRITERION FOR VALIDATION RUNS	INPUT VALUE FOR VALIDATION RUNS
ARSPED	the aircraft are assumed to be F-16's	13350 m/minutes (500 mph)
AO	second echelon movement is assumed to start in the daylight hours	50 meters
AODA	(REF 7: 2-30)	50 meters
AONI	(REF 7: 2-30)	30 meters
CEP	the actual CEP for F-16's delivering cluster-bomb units is 50 meters, the model uses the assumption that all the weapons delivered will land within 100 meters of the target	100 meters
CLVEH	assumption	2
DAYBRK	the model considers the time to change from total darkness to total daylight to be 1 hour	60 minutes
DAYTIM	winter, eight hours of full daylight	480 minutes
NTFALL		60 minutes

NTRUCK	the value used was chosen to allow a long enough time for a delay (if existent) to be apparent in the final results	2000
NUMPLA	chosen for validation purposes	100
PKAAA		.025
PKSAM	varied over a range of values	****
PKTRUC	varied over a range of values	****
VO	second echelon movement is assumed to start in the daylight hours	500 meters
VODA	(REF 1: 2-31)	500 m/minute
VONI	(REF 1: 31) (REF 2: 2-31)	250 m/minute

To initialize the subroutine which updates the spacings and velocity, the value is NXEV(J), for some J, must be 5, 6, 7 or 8. However, since some plane has the type of its next event in that attribute for J≤NUMPLA, the indices for the events which update the spacing and velocities must be greater than the number of planes. (For example, if the user of the program wishes the change from night to day to happen at time 80, the value in NXEVT(NUMPLA + 1) would be 80 and NXEV (NUMPLA + 1) would be 5.) The program has the capability built in to further update any changes of the day/night type and no further input for this type of change is needed.

For the changes in the actual meteorological environment, the time and type of each change must be input by the user. If the model begins in permissive type weather, the start times of the non-permissive weather to be encountered throughout the simulation are stored in the NUMPLA + 2, 4, 6, 8... n positions

of NXEVT time with the value 7 in the associated cells in NXEV. The end times of the non-permissive weather are stored in the NUMPLA + 3, 5, 7, 9... n+1 positions in NXEVT with the value 8 in the associated cells of NXEV. Since weather is cyclic, the values in the NUMPLA = 2, 3, 4, 5, 6... n+1 cells of NXEVT must be input to insure the model realistically simulates the environment under consideration.

For the validation runs the following values were used to initiate changes due to weather:

CELL	NXEVT VALUE	NXEV VALUE	MEANING
101	100	5	At time 100, the model will simulate the change in activity, due to sunset.
102	80	7	At time 80, non-permissive weather will set in.
103	90	8	At time 90, permissive weather will resume.

Verification Runs and Results

To insure the model moved trucks through the second echelon as expected, a run was accomplished where trucks move toward the FEBA without hinderance from weather changes or attack. The model yielded a time of 126 when all trucks had completed movement. This was the expected results because under the conditions used, the model simply becomes a movement of 667 trucks through each channel. Since 10 trucks enter a channel per minute when unhindered, the last trucks will enter the channels after 66 minutes. Given it takes 60 minutes for a truck to traverse a channel, the expected time second

echelon movement is to be completed is 126.

Once verification of the truck movement was accomplished, the model was checked to see if the attack on the second echelon had the intended effect. Since the purpose of the attack on the second echelon was to not only kill trucks, but to also delay their arrival to the FEBA, when airplanes are allowed to attack the channels, the time it takes for all trucks to reach the FEBA should increase from 126. Using a PKSAM of .5 and a PKTRUC of .3, 10 runs of the simulation using different random numbers yielded a mean time of 141 for all trucks to reach the FEBA. Therefore, attacks were having a delaying effect on second echelon movement.

As the variables PKAAA and PKTRUC are varied, there should be predictable effects on the model. If PKSAM is increased, the amount of time it takes for second echelon movement to be completed should decrease because the number of planes surviving long enough to attack will decrease. Conversely, as PKTRUC rises, there will be more trucks killed and, therefore, more time spent on damage control. Hence, all else remaining the same, a rise in PKTRUC should cause slower second echelon movement.

Appendix D contains a table presenting the average amount of time (for 10 runs) it took for second echelon movement to be completed given certain values of PKSAM and PKTRUC. For each value of PKSAM, as PKTRUC increased, the time it took for second echelon movement to finish also increased. Also, for each value of PKTRUC, as PKSAM increased, the average time

decreased. A two-way ANOVA was run using the values in the table. The results of the test are given beneath the table. Comparing the two F statistics with $F_{2,\ 4,\ .05}$ shows that the null hypotheses that there is no difference between the rows or between the columns can both be rejected. Therefore, it can be said that at the .95 level of confidence that the values PKSAM and PKTRUC make a significant difference in the outcome of the model. This shows the simulation is yielding results which are consistent with expected outcomes.

In the final step for verification, several runs were made of the simulation which yielded the status variables for the program as output. In addition, at certain times, the values in the array NXEVT were checked to see if the model chose the correct next event. Also inspected was the spacing between vehicles in the channels. In every instance, the variables checked had values consistent with the intentions of the author. From this analysis, it was concluded that the model was simulating second echelon interdiction adequately under the assumptions made.

V The Comparison of DCUBE and the Simulation

The comparison of the results of the simulation and the results of the DCUBE required manipulation of the output from the simulation. The probability of kill input to the DCUBE was defined as the percentage of trucks in a channel killed by a single aircraft. To arrive at this number from the simulation, the total number of vehicles in a channel at the time an attack was made by an airplane was calculated for each attack. Then, the sum of all those numbers was taken. Dividing that number into the number of trucks killed by aircraft yields a PK corresponding to the DCUBE's PK for trucks. In addition, since there are two ways an aircraft can be killed in the simulation, an overall probability of kill for the airplanes is calculated by dividing the total number of attacks on airplanes into the number of aircraft killed.

For the purpose of comparison to the DCUBE model, all the inputs for the simulation runs remained the same as for validation, except for the number of trucks, which was raised to 5000. Using those inputs, three different sets of values for PKSAM and PKTRUC were used to generate data. For each case, the table below gives the mean time (from 10 repetitions) for all trucks to reach the FEBA, the 95% confidence interval for the mean, the PKT (probability of kill for trucks corresponding to DCUBE) generated by the simulation, and the PKA (probability of kill for airplanes corresponding to DCUBE) generated by the simulation.

		Case	
	1	2	3
PKSAM	.1	.05	.15
PKTRUC	.5	.4	.5
MEAN	201	210	186
CI UL	230	245	362
CI LL	172	75	10
PKT	.008	.01	.01
PKA	.17	.09	.29

* Due to the large variance within case three, it will not be considered in the comparison of the two models.

Figure 2 Results of the Simulation

The next step for the comparison of the two models is to input the probability of kill for trucks (PKT in table) and the probability of kill for airplanes (PKA in table) into the DCUBE model and calculate the results. (Since the simulation did not consider channel cuts, the number of planes assigned to indirect attacks is set to zero for DCUBE.) For case 1, the DCUBE analytically calculated the time 174.6 for the arrival of all trucks to the FEBA. Case 2 yielded a time of 179.3. Since both values lie within the 95% confidence interval for the mean of the respective simulation, it cannot be said that the two models are significantly different. However, this may be misleading. Note that in both cases, the result from the DCUBE lies very close to the lower limit of the confidence

interval. After calculating the results of the DCUBE manually, this trend is apparently due to the difference in the way the delay is calculated in the two models. Recall that the DCUBE assumes that all vehicles in a channel are able to participate in damage control. This says that even those vehicles ahead of damage points will participate. The simulation, on the other hand, assumes that only those vehicles within 1000 meters behind a damage point will participate in damage control. Any vehicles ahead of a damage point will move on toward the FEBA. Therefore, since the DCUBE assumes more vehicles will assist in damage control, the delay associated with killed vehicles is much smaller, and vehicles will therefore reach the FEBA sooner. (It is not readily apparent which assumption is more valid, there are merits to both. A potential solution to the problem will be discussed in the final chapter.)

This difference in time for the completion of damage control causes even greater problems when the number of vehicles reaching the FEBA per unit time is considered. Listed below are the number of vehicles which have reached the FEBA at the end of each half hour for the two models using case one's results.

	DCUBE	SIMULATION
$\frac{1}{2}$ hour	790	714
1 hour	1585	1422

	DCUBE	SIMULATION
$1\frac{1}{2}$ hours	2392	2016
2 hours	3204	2664
$2\frac{1}{2}$ hours	4024	3296
3 hours	4859	4049
$3\frac{1}{2}$ hours		4690

Figure 3 Arrival Rates Given By Both Models

Due to the lower delay time associated with each attack, vehicles in the DCUBE model are able to move to the FEBA at a faster average speed. This means that vehicles will not be subjected to as many attacks and, therefore, less vehicles will be killed. In addition to this, the number of vehicles able to reach the FEBA per unit time increases with time in the DCUBE and decreases (generally) with time in the simulation. These differences further necessitate the removal of the ambiguity from the two model's delay component.

If the delay was calculated in a manner so as to yield comparable delay times in both models, the time the DCUBE yields for completion of second echelon movement should approach the mean for the simulation runs. This, coupled with the examination of the process involved with the DCUBE yields the conclusion that the difference between the two models can be reconciled and both models can be used as tools for analysis of second echelon air interdiction.

VI Conclusion

The purpose of this thesis was to develop a simulation model against which the DCUBE model could be compared. Although the results of the simulation and the DCUBE model were not shown to be significantly different, the simulation model does have its place in the analysis of second echelon air interdiction. This is due to the fact that the DCUBE is unable to incorporate the variance in the problem caused by human and mechanical factors. Since the purpose of models such as those presented in this report is to give the decision maker some input into the choice between, for example, weapons systems, the variance within the results of a test can play a key part in determining whether or not a significant difference exists between alternatives. The simulation model, given the time and money to complete enough repetitions, is able to incorporate variance and, therefore, able to yield results which may lead to different conclusions than DCUBE.

The simulation model is able to incorporate the variance within the problem due to the fundamental differences between an analytic model and a simulation. At different points in Chapter II, points which were potential weaknesses with the DCUBE were brought out. Among these were the constant kill rates for both planes and trucks assumed by DCUBE. The simulation determines the kill rates through the use of a Monte Carlo process which, despite the assumption of constant PK's, removed the constant kill rate from the model. This provides

a more realistic attrition rate for airplanes and trucks, both of which can be key parts for an analysis. In addition to the problem with the PK's, the method the DCUBE uses to calculate the delay associated with killed vehicles is questionable. Assuming that all survivors in a channel will participate in damage control is unrealistic. Only those vehicles within a certain distance behind a damage point will assist in road clearance and the simulation incorporates this fact.

In conclusion, there are several key differences between the simulation and DCUBE. (Figure 4 provides a summary of the differences.) Due to these differences, the simulation provides an alternative to DCUBE for the analyst in determining the effects of second echelon air interdiction. Although the numbers from neither of the models provides the actual numbers which can be expected in the event of a NATO - Warsaw Pact conflict, they do provide a basis for an adequate comparison of results when inputs into the models are varied.

DIFFERENCE IN	DCUBE	SIMULATION
PK's	assumes constant kill rates for trucks and planes uses an exponential approximation to binomial	simulates attacks and determines kill rates through Monte Carlo process uses binomial
DELAY	assumes all vehicles in a channel will participate in damage control	assumes only those vehicles directly behind a kill will participate in damage control
AAA	allows aircraft attrition only at prescribed times	allows aircraft attrition at several times during a flight
TRUCK MOVEMENT	either all vehicles move or none move	allows movement to take place for trucks ahead of all damage points
CUTS	incorporates channel cuts	does not incorporate channel cuts
COST	1 run yields results	many repititions required

Figure 4
General Comparison of the Simulation and DCUBE

Recommendation

This report shows that further work must be done in researching the scenario modeled by DCUBE and the simulation. The two areas which necessitate immediate attention are the delay components and the attacks on the channel themselves. These two "ambiguities" take away from the model in their present forms.

The simulation did not incorporate channel cuts because any estimate of the locations of choke points would have been pure speculation. Current intelligence yields the fact that three to five is the actual number of dependable roadways the Warsaw Pact will use to reinforce their front line troops, but unclassified reports do not mention where along these channels lie the critical areas. The actual location and number of choke points are critical because of the delay associated with a successful attack.

More research should be done in the literature to determine if any rational choice of choke points could be incorporated into the simulation. They could be put into the simulation's search and attack routines by use of indicator variables. From their insertion, the vehicles would experience a greater delay on the way to the FEBA and more trucks could be killed because they are in the channels longer if choke points are destroyed.

The DCUBE's method of assuming a constant number of cuts per sortie does not seem realistic. The variance in weapon accuracy and effectiveness will cause the number to vary.

In addition, there are only a finite number of choke points which may be attacked. Therefore, the results of the literature search should be incorporated into both the DCUBE and simulation and a better estimate of vehicle delay calculated.

It is apparent that the difference in the two delay components caused the times from the DCUBE model to fall toward the lower limits of the confidence intervals from the simulation. In the present theater war games run by different War Colleges, the clearance of killed vehicles should play an important role. Investigation into the assumptions made in these games should yield a better method of calculating delay. In addition, investigation into data from the Vietnamese Conflict could indicate the most recent procedures employed to remove damaged vehicles from vital roadways. Despite the fact that the techniques used were comparatively primative in nature, some general method of clearing damage may be found. One of these two sources should yield a more realistic method of calculating the delay that both models could use.

Future Research Topics

To further develop the results of the simulation, the model could be extended to both sides interdicting the other's supply lines. In addition, more realistic values for the numbers of trucks and airplanes could be input into the model and the result input into the Lanchester equations. In the runs used in this thesis, the numbers chosen were strictly for validation purposes and not large enough to allow the Lanchester

equations to be used. Choosing larger values would have slowed computer turn around time to a point where meaningful post-run analysis would not have been possible. However, the model is verified and future students could use it to investigate the effects of interdiction on reinforcement efforts over a longer period of time.

The investigation of the literature to find realistic chokes point and delay components could be the basis of a future research topic. This literature search would be extensive and the results of the search could be incorporated into both the simulation and DCUBE models. Additional verification and statistical analysis would be required, but the times both models yield should converge to a common number and the full simulation would then be in a form which could be employed by the Air Force in conjunction with the DCUBE model.

Bibliography

1. Bogusch, Jack; Major USAF.

2. Friedman, Frank L. et al. Problem Solving And Structured Programming In Fortran. Reading, Massachusetts: Addison-Wesley Publishing Co., 1977.

3. Harris, Kathleen. Tactical Airpower in NATO Contingencies: A Joint Air-Battle/Ground-Battle Model (TALLY/TOTEM). Santa Monica, CA: RAND Corp., 1974.

4. Mendenhall, William et al. Mathematical Statistics With Applications. North Scituate, Massachusetts: Duxbury Press, 1973.

5. Selected Soviet Weapons and Equipment. Fort Knox, Kentucky: US Army Armor School, 1976.

6. Shannon, Robert. Systems Simulation, The Art and Science. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1975.

7. Soviet Army Operations. Arlington, VA: Department of the Army, 1978.

8. Strauss, Walter J. et al. The DCUBE Model. Chicago: A.T. Kearney, Inc., 1980.

APPENDIX A

SUMMARY OF DEFINITIONS

A_j	-	Average spacing between consecutive targets in the channels during time interval j (meters).
B_{10}	-	Number of aircraft initially assigned to direct attacks.
B_{20}	-	Number of aircraft initially assigned to indirect attacks.
C	-	Number of distinct channel cuts per sortie.
C_D	-	Vehicle-hours required for damage control per dead vehicle (hrs).
D	-	Duration of second echelon movement--time required until the sum of those targets which have reached the FEBA and those which have been killed is equal to the total number of targets, T (hrs).
E_j	-	Probability that a given aircraft survives to attack during time interval j.
K	-	Number of targets killed per sortie in an air attack against a unit of size N_0.
N_j	-	Number of surviving targets in a channel during time interval j.
P_A	-	Probability that a given sortie fails to survive long enough to attack its target.
P_{Dj}	-	Probability that the targets in the channel experience a delay at any given time during time interval j.
P_{Kj}	-	Probability that a given target is killed during time interval j.
P_{Uj}	-	Probability that a given dead vehicle has not yet been cleared (i.e., subjected to damage control) during time interval j.
Q	-	Number of parallel channels along which the reinforcements may flow.
R_j	-	Rate at which targets arrive at the FEBA per channel during time interval j(1/hrs).

T - Number of reinforcements which may flow to the FEBA.

U - Number of sorties per aircraft per hour (1/hrs).

V_j - Average speed of the targets during time interval j (km/hr).

W - Length of time channel is cut per cut (hrs).

X_j - Net gain in targets between interval (j-1) and the start of interval j. This net gain is taken before the air attack occurs in interval j and does not subtract those vehicles killed in interval j.

Y_j - Number of killed targets that have not yet been cleared (i.e., subjected to damage control) immediately following the air attack in interval j.

Z_j - Number of targets having arrived at the FEBA from all Q channels by the end of interval j. (Note, an unsubscripted Z is used to mean the number of reinforcement vehicles having reached the FEBA by the time that second echelon movement ceases.)

dt - Duration of each time interval (hrs).

dT_j - Expected time required to clear the channel as of the end of the air attack in interval j (hrs).

dT_{1j} - Expected time required to reform damage control upon all killed vehicles as of the end of the air attack in interval j (hrs).

dT_{2j} - Expected time required to repair cuts in the channel as of the end of the air attack in interval j (hrs).

APPENDIX B

MATHEMATIC FORMULATION OF ARRIVAL RATE MODEL

Inputs: $A_O, B_{10}, B_{20}, C, C_D, K, N_O, P_1, Q, T, U, V_O, W, Dt$

1) $\text{alph} = C \cdot W \cdot B_{20} \cdot \frac{U}{Q}$ — used for computational purposes

2) $T_o = N_o \cdot \frac{A_o}{V_o}$

3) $E_1 = \exp(-U \cdot P_A \cdot dt)$ — probability that an aircraft attacks during the first interval

4) $\text{beta} = \frac{K \cdot B_{10} \cdot U \cdot A_o}{Q \cdot V_o}$ — used for computational purposes

5) $j = 1$

6) $X_1 = 0$

7) $P_{K1} = 1 - \exp(-B \cdot E_1 \cdot \frac{dt}{T_o})$ — probability that a given target is killed during interval 1

8) $Y_1 = P_{K1} \cdot N_o$ — computes the number of second echelon vehicles killed during interval 1

9) $F_{K1} = Q \cdot Y_1$ — gives the total number of second echelon vehicles killed during interval 1

10) $N_1 = N_o - Y_1$ — computes the total number vehicles remaining

11) $dT_{11} = \frac{C_D \cdot P_{K1}}{1-P_{K1}}$ — amount of delay during interval 1 coming from direct attack

12) $dT_{21} = alph \cdot E_1 \cdot dt$ — amount of delay during interval 1 coming from indirect attack

13) $dT_1 = dT_{11} + dT_{21}$ — total delay during first interval

14) $P_{D1} = 1- \exp(-\frac{dT_1}{dt})$ — probability a vehicle experiences a delay during interval 1

15) $R_1 = (1-P_{D1}) \cdot \frac{N_1}{T_o}$ — rate of arrivals during first interval

16) $Z_1 = Q \cdot R_1 \cdot dt$ — total number of arrivals to the FEBA during the first interval

17) $j = j + 1$ — beginning of interval loop

18) $X_j = (\frac{N_o}{N_{j-1}} - 1) \cdot R_{j-1} \cdot dt$ — computes the net gain in targets from interval j-1 to interval j

19) $P_{Kj} = 1- \exp(-beta \cdot E_1^j \cdot \frac{dt}{T_o})$ — probability that a given target is killed during interval j

20) $P_{Uj} = \dfrac{dT_{1j-1} - (1-\exp(-\frac{dT_{1j-1}}{dt}))dt}{dT_{1j-1}}$ — probability a vehicle killed during j-1 is still blocking a channel

21) $F_{Kj} = F_{Kj-1} + Q \cdot P_{Kj} \cdot (N_{j-1} + X_j)$ — computes the total number of vehicles killed during all intervals

22) $Y_j = P_{Kj} \cdot (N_{j-1} + X_j) + Y_{j-1} \cdot P_{Uj}$ — computes the total number of killed vehicles blocking a channel during interval j

23)	$N_j=(N_{j-1}+X_j)(1-P_{Kj})$	number of vehicles remaining after the air attack during interval j
24)	$dT_{2j}=alph\cdot E_1^j\cdot dt + (dT_{2j-1} - (1-\exp(-\frac{dT_{2j-1}}{dt})).dt)$	delay coming from indirect attack
25)	$dT_j = dT_{1j} + dT_{2j}$	total delay during j
26)	$P_{Dj} = 1-\exp(-\frac{dT_j}{dt})$	probability a vehicle is delayed during j
27)	$R_j = (1-P_{Dj})\cdot\frac{N_j}{T_0}$	rate vehicles arrive at the FEBA during j
28)	$Z_j = Z_{j-1} + Q\cdot R_j\cdot dt$	total arrivals to the FEBA through j
29)	$Z_j + F_{Kj} < T$ yes: go to 17 no: continue	this step checks whether the sum of the number of killed vehicles (F_{Kj}) and the number of vehicles which have reached the (FEBA) exceeds the total number of enemy vehicles (T)
30)	$lamda=\frac{Z_j+F_{Kj}-T}{Q\cdot((R_j\cdot dt)+(P_{Kj}\cdot(N_{j-1}+X_j)))}$	computes the portion of the last interval during which there were no arrivals at the FEBA
31)	$D=(j-lamda)dt$	calculates the total duration of second echelon movement
32)	$Z = Z_{j-1}+Q(1-lamda)\cdot R_j\cdot dt$	calculates the total number of vehicles which reached the FEBA

34) $F_K = T - Z$ calculates the total number of vehicles which were killed during second echelon movement

Outputs(for all j): R_j, dT_j, F_{Kj}

APPENDIX C

```
THIS IS THE TYPE OF ALL SIMULATION VARIABLES

      REAL NXEVT(NUMPLA+5),NXTEVT,PKSAM,FRWARD,TRUCK(NTRUCK)
      REAL DAMPT(3,150),CEP,ANGLE,DIST,FRONT,BACK
      REAL PKTRUC,POINT,CLVEH,PKAA(2),CLDAM(3,150)
      REAL ARSPED,TRAVT,TMOVE,VO,AO,ATTACK,CNTROD
      REAL REPTIM,TPLACE,PKAAA
      REAL LSTEVT
      INTEGER NUMPLA,NXEV(NUMPLA+5)
      INTEGER PLNCOL(NUMPLA+5),PLNROW(NUMPLA+5)
      INTEGER NWEP(NUMPLA+5),LEFT,KILL,NUMDAM(3),Q,H
      INTEGER LOWJ,NTRUCK,TRUCR(NTRUCK),NUMBER,WAIT
      INTEGER NTRUC(3,30)
      INTEGER DAM(3,30)
      INTEGER LAG,XX,NTHRES,TRAAA(NTRUCK)
      DO 10 J=1,NUMPLA
      NXEV(J)=3
      NWEP(J)=1
      L=J/2
      Z=L
      IF(MOD(J,2).NE.0.0) GO TO 9
      X=RANF()
      NXEVT(J)=Z+FLYTIM + X/1000
      NXEVT(J-1)=NXEVT(J)
 9     CONTINUE
10     CONTINUE
  11 LSTEVT=NXTEVT
      L=0
      K=0
      NXTEVT=1000000
      DO 20 J=1,NUMPLA
      IF(NXEVT(J).GT.NXTEVT) GO TO 19
      IF(NXEVT(J).EQ.NXTEVT) GO TO 18
      NXTEVT=NXEVT(J)
      L=0
      K=J
      GO TO 19
18    CONTINUE
      L=J
19    CONTINUE
20    CONTINUE
      IF(NXEV(K).EQ.1) GO TO 5000
      IF(NXEV(K).EQ.2) GO TO 6000
      IF(NXEV(K).EQ.3) GO TO 2000
      GO TO 7000
```

THIS IS THE ROUTINE THAT SIMULATES SAM ATTACK AT THE FEBA

```
2000 CONTINUE
     IF(NWEP(K).EQ.0) GO TO 145
     LEFT=0
     DO 130,I=1,2
     X=RANF()
     IF(X.LT.PKSAM) GO TO 130
     LEFT=LEFT+1
130  CONTINUE
     IF(LEFT.NE.2) GO TO 140
     NXEVT(K)=NXEVT(K)+15000/ARSPED
     NXEVT(L)=NXEVT(K)
     NXEV(K)=2
     NXEV(L)=2
     GO TO  151
140  CONTINUE
141  NXEVT(K)=NXEVT(K)+FLYTIM
     NXEV(K)=1
     NXEVT(L)=1000001
     NXEV(L)=100
     IF(LEFT.NE.0) GO TO  151
     NXEVT(K)=1000001
     NXEV(K)=100
     GO TO  151
145  CONTINUE
     DO 150 I=1,2
     N=L
     IF(I.EQ.1)N=K
     X=RANF()
     IF(X.LT.PKSAM) GO TO  148
     NXEV(N)=1
     NXEVT(N)=NXEVT(N)+FLYTIM
     GO TO  149
148  CONTINUE
     NXEVT(N)=1000001
     NXEV(N)=100
149  CONTINUE
     IF(L.EQ.0) GO TO  151
150  CONTINUE
151  CONTINUE
     NPLATT=NPLATT+2
     IF(L.EQ.0) NPLATT=NPLATT-1
     GO TO 4000
```

THIS ROUTINE SIMULATES THE ATTACK ON THE SECOND ECHELON

```
3000 CONTINUE
     MKL=0
     NTATT=0
     DO 3066 I=1,30
     NTATT=NTATT+NTRUC(PLNROW(K),I)
3066  CONTINUE
     KILL=0
     FRWARD=0.0
     IF(PLNCOL(K).EQ.30) GO TO 61
     IF(NUMDAM(PLNROW(K)).EQ.0) GO TO 51
     IF(DAMPT(PLNROW(K),1).LT.(PLNCOL(K)*1000)) GO TO 51
     DO 50 I=1,NUMDAM(PLNROW(K))
     IF(DAMPT(PLNROW(K),I).LT.PLNCOL(K)*1000) GO TO 62
     FRWARD=DAMPT(PLNROW(K),I)
50   CONTINUE
     GO TO 62
51   CONTINUE
     DO 60 J=LOWJ,NTRUCK
     IF(TRUCR(J).NE.PLNROW(K)) GO TO 59
     IF(TRUCK(J).GE.PLNCOL(K)*1000+1000) GO TO 59
     IF(TRUCK(J).LT.PLNCOL(K)*1000-1000) GO TO 59
     FRWARD=TRUCK(J)
     GO TO 62
59   CONTINUE
60   CONTINUE
61   CONTINUE
     FRWARD=29999
62   CONTINUE
     ATTACK=FRWARD-500
     DO 80 LL=1,2
     X=RANF()
     Y=RANF()
     Z=RANF()
     IF(Y.GT..5) CNTROD=ATTACK+X*CEP
     IF(Y.LE..5) CNTROD=ATTACK-X*CEP
     ANGLE=Z*(3.1416/2)
     IF(ANGLE.LT..4625) DIST=200/(SIN((1.5707-ANGLE)))
     IF(ANGLE.GE..4625) DIST=100/(SIN(ANGLE))
     FRONT=CNTROD+DIST
     BACK=CNTROD-DIST
     DO 90 J=LOWJ,NTRUCK
     IF (TRUCR(J).NE.PLNROW(K)) GO TO 89
     IF(TRUCK(J).GT.FRONT) GO TO 89
     IF(TRUCK(J).LT.BACK) GO TO  80
     X=RANF()
     IF(X.GT.PKTRUC) GO TO 89
     MKL=MKL+1
     IF(KILL.EQ.0) POINT=TRUCK(J)
```

```
      MKL=MKL+1
      NUMDAM(PLNROW(K))=NUMDAM(PLNROW(K))+1
      DO 1000 I=1,NUMDAM(PLNROW(K))
      IF(I.EQ.NUMDAM(PLNROW(K))) GO TO 990
      IF(DAMPT(PLNROW(K),NUMDAM(PLNROW(K))-I).GT.TRUCK(J))
     *GO TO 990
      DAMPT(PLNROW(K),NUMDAM(PLNROW(K))-(I-1))=DAMPT(PLNROW(K)
     *,NUMDAM(PLNROW(K))-I)
      CLDAM(PLNROW(K),NUMDAM(PLNROW(K))-(I-1))=CLDAM(PLNROW(K)
     *,NUMDAM(PLNROW(K))-I)
      GO TO 1000
990   DAMPT(PLNROW(K),NUMDAM(PLNROW(K))-(I-1))=TRUCK(J)
      IF(KILL.EQ.0) JJ=NUMDAM(PLNROW(K))-(I-1)
      GO TO 1010
1000  CONTINUE
1010  CONTINUE
      TRUCK(J)=40000
      KILL=KILL+1
      KILLS=KILLS+1
      TRUCR(J)=4
89    CONTINUE
90    CONTINUE
80      CONTINUE
      SURV=0
      DO 1100 J=LOWJ,NTRUCK
      IF(TRUCR(J).NE.PLNROW(K)) GO TO 1100
      IF(TRUCK(J).GT.POINT) GO TO 1100
      IF(TRUCK(J).LT.(POINT-1000)) GO TO 1110
      SURV=SURV+1.0
1100   CONTINUE
1110   CONTINUE
      REPTIM=MKL*CLVEH/SURV
1111    CONTINUE
      DO 1200 Q=JJ,JJ+KILL
      CLDAM(PLNROW(K),Q)=NXTEVT+REPTIM
1200  CONTINUE
      NPLATT=NPLATT+2
      NTTLAT=NTTLAT+NTATT
      KLLTTL=KLLTTL+NUMDAM(PLNROW(K))
      NUMBER=0
      DO 100 J=LOWJ,NTRUCK
      IF(TRUCR(J).NE.PLNROW(K)) GO TO 99
      IF(TRUCK(J).GT.ATTACK+500) GO TO 99
      IF(TRUCK(J).LT.ATTACK-500) GO TO 99
      IF(TRAAA(J).EQ.0) GO TO 99
      NUMBER=NUMBER+1
99    CONTINUE
100   CONTINUE
      IF(MOD(NUMBER,2).EQ.0) GO TO 105
      A=NUMBER/2+.5
```

```
      B=A-1
      GO TO 106
 105  CONTINUE
      A=NUMBER/2
      B=A
106   CONTINUE
      PKAA(1)=1-((1-PKAAA)**A)
      PKAA(2)=1-((1-PKAAA)**B)
      DO 110 I=1,2
      N=L
      IF(I.EQ.1)N=K
      NWEP(N)=0
      X=RANF()
      IF(X.GT.PKAA(I)) GO TO 108
      PLNROW(N)=4
      NXEVT(N)=1000001
      GO TO 109
108   CONTINUE
      NXEV(N)=3
      NXEVT(N)=NXEVT(N)+(((30-PLNCOL(N))*1000)+15000)/ARSPED
109   CONTINUE
110   CONTINUE
      GO TO 4000
```

THIS IS THE ROUTINE THAT HANDLES AIRCRAFT TURNAROUND

```
5000 CONTINUE
160  IF(L.EQ.0) GO TO 161
     NXEVT(K)=NXEVT(K) + FLYTIM + 25.00001
     NXEVT(L)=NXEVT(K)
     NXEV(K)=3
     NXEV(L)=3
     NWEP(K)=1
     NWEP(L)=1
     GO TO 163
161   IF (WAIT.EQ.0) GO TO 162
     L=LAG
     WAIT=0
     GO TO 160
162  LAG=K
     NXEVT(K)=1111111
163  CONTINUE
     PLNROW(K)=0
     PLNROW(L)=0
     GO TO 4000
```

```
 THIS ROUTINE SIMULATES THE SEARCH FOR TARGETS

6000 CONTINUE
     IF(PLNROW(K).NE.0) GO TO 39
     PLNCOL(K)=30
     PLNCOL(L)=30
     NMDMPT=NUMDAM(1)+NUMDAM(2)+NUMDAM(3)
     IF(NMDMPT.NE.0) GO TO 38
37    CONTINUE
     X=RANF()
     XX=INT((X*3)+1)
     PLNROW(K)=XX
     PLNROW(L)=PLNROW(K)
     GO TO 39
38    CONTINUE
     DO 40 H=1,3
     IF(NUMDAM(H).NE.0) GO TO 40
     PLNROW(K)=H
     PLNROW(L)=PLNROW(K)
40   CONTINUE
39   CONTINUE
     IF(PLNROW(K).EQ.0) GO TO 37
     IF(DAM(PLNROW(K),PLNCOL(K)).EQ.1) GO TO 41
     IF(NTRUC(PLNROW(K),PLNCOL(K)).LT.NTHRES) GO TO 41
     NXEV(K)=4
     NXEV(L)=4
     GO TO 42
41    CONTINUE
     NXEV(K)=2
     NXEV(L)=2
     NXEVT(K)=NXEVT(K)+1000/ARSPED
     NXEVT(L)=NXEVT(K)
     PLNCOL(K)=PLNCOL(K)-1
     PLNCOL(L)=PLNCOL(K)
     IF(PLNCOL(K).NE.0) GO TO 42
     NXEV(K)=3
     NXEV(L)=3
     NXEVT(L)= NXEVT(L)+ 45000/ARSPED
     NXEVT(K)=NXEVT(L)
     NWEP(K)=0
     NWEP(L)=0
42   CONTINUE
     IF(NXEV(K).EQ.4) GO TO 3000
     GO TO 4000
```

THIS ROUTINE HANDLES WEATHER EFFECTS

```
7000 CONTINUE
     IF(NXEV(K).NE.5) GO TO 7001
     III=III+1
     PKAAA=PKAAA-(((PKAADA-PKAANI)*.2)*1/NN)
     VO=VONI
     AO=AONI
     NXEV(K)=NXEV(K)+NTFALL/5
     IF(III.NE.5) GO TO 4000
     III=0
     NXEV(K)=6
     NXEVT(K)=NXEVT(K)+1440-NTFALL-DAYBRK-DAYTIM
     NTHRES=3
     GO TO 4000
7001 IF(NXEV(K).NE.6) GO TO 7002
     III=III+1
     PKAAA=PKAAA+(((PKAADA - PKAANI)*.2)*1/NN)
     VO=VODA
     AO=AODA
     NXEVT(K)=NXEVT(K) + DAYBRK/5
     IF(III.NE.5) GO TO 4000
     NXEVT(K)=NXEVT(K) + DAYTIM
     NTHRES=3
     III=0
     NXEV(K)=5
     GO TO 4000
7002 IF(NXEV(K).NE.7) GO TO 7003
     NXEVT(K)=1000001
     PKAAA=PKAAA/2
     NN=2
     NTHRES=3
     NXEV(K)=8
     AO=AONI
     VO=VONI
     GO TO 4000
7003 PKAAA=PKAAA*2
     NXEVT(K)=1000001
     AO=AODA
     VO=VODA
     NN=1
     NTHRES=3
```

```
 THIS ROUTINE HANDLES TRUCK MOVEMENT

4000 CONTINUE
     DO 1011 H=1,3
     DO 1020 J=1,30
     NTRUC(H,J)=0
1020  CONTINUE
1011  CONTINUE
    *'LAST EVENT TIME IS',2X,F8.4)
     TRAVT=NXTEVT-LSTEVT
     TMOVE=VO*TRAVT
     DO 300 H=1,3
     N=NUMDAM(H)
     IF(N.EQ.0) THEN
     N=1
     GO TO 710
     ENDIF
     DO 400 I=1,NUMDAM(H)
     IF(CLDAM(H,I).GT.NXTEVT) GO TO 400
     DAMPT(H,I)=0
     N=N-1
400  CONTINUE
     IF(N.EQ.NUMDAM(H)) GO TO 710
     DO 500 I=1,N
450  IF(DAMPT(H,I).NE.0) GO TO 610
     DO 600 J=I,NUMDAM(H)
     DAMPT(H,J)=DAMPT(H,J+1)
     CLDAM(H,J)=CLDAM(H,J+1)
600  CONTINUE
     GO TO 450
610  CONTINUE
500  CONTINUE
     DO 700 J=N+1,NUMDAM(H)
     DAMPT(H,J)=0
     CLDAM(H,J)=0
700  CONTINUE
     NUMDAM(H)=N
710  CONTINUE
     DO 750 I=1,NUMDAM(H)+1
     IF(I.NE.1) GO TO 760
     FRONT=30000
     BACK=DAMPT(H,1)
     IF(NUMDAM(H).EQ.0) BACK=0-1
     GO TO 770
760  CONTINUE
     FRONT=DAMPT(H,I-1)
     BACK=DAMPT(H,I)
     IF(I.EQ.NUMDAM(H)) BACK=0-1
     IF(FRONT.GT.BACK) GO TO 769
     DO 9509 II=1,NUMDAM(H)+5
```

```
9509  CONTINUE
769      CONTINUE
770  CONTINUE
 771   DO 800 J=LOWJ,NTRUCK
     IF(TRUCK(J).EQ.0) TRUCR(J)=H
     IF(TRUCR(J).NE.H) GO TO 780
     IF(TRUCK(J).GT.FRONT) GO TO 780
     IF(TRUCK(J).LT.BACK) GO TO 810
     FRONT=FRONT-AO
     IF(FRONT.EQ.(30000-AO)) FRONT=30000
     IF(FRONT.LT.BACK) GO TO 810
     IF  (FRONT.LT.TRUCK(J)) FRONT=TRUCK(J)
     TRUCK(J)=MIN(TPLACE,FRONT)
     FRONT=TRUCK(J)
780  CONTINUE
800  CONTINUE
810  CONTINUE
     IF(NUMDAM(H).EQ.0) GO TO 300
750  CONTINUE
300  CONTINUE
     DO 7530 J=1,NTRUCK
     IF(TRUCK(J).GE.30000) GO TO 7530
     IF(TRUCK(J).LE.0) GO TO 7530
     KKK=1+ TRUCK(J)/1000
     NTRUC(TRUCR(J),KKK)=NTRUC(TRUCR(J),KKK)+1
7530  CONTINUE
     DO 7550 H=1,3
     DO 7540 J=1,NUMDAM(H)
     KKK=1+DAMPT(H,J)/1000
     DAM(H,KKK)=1
7540   CONTINUE
     DO 7545 J=1,30
7545  CONTINUE
7550   CONTINUE
     MMM=0
     DO 7510 J=1,NTRUCK
     IF(TRUCK(J).GE.30000) MMM=MMM+1
7510  CONTINUE
     DO 7990 J=1,NTRUCK
     IF(TRUCK(J).LE.0) GO TO 7991
7990  CONTINUE
7991  CONTINUE
     LMN=MMM-KILLS
     IF(NXTEVT.GE.NNN) THEN
     NNN=INT(NXTEVT)+1
     ENDIF
     IF(LMN.LT.1) THEN
     KLLTTL=0
     NTTLAT=0
     NPLATT=0
```

```
      KLPL=0
      ENDIF
      IF(MMM.LT.NTRUCK) GO TO 11
      DO 7995 I=1,100
      IF(NXEVT(I).GE.1000000) KLPL=KLPL+1
7995    CONTINUE
      STOP
      END
```

APPENDIX D

This chart gives the mean time second echelon movement ended for ten repetitions while varying the inputs in the manner indicated on the chart.

PKTRUC

PKSAM	.3	.4	.5
.025	140.6	150.3	158.0
.05	137.3	144.5	151.0
.075	135.5	138.0	142.7

Figure D-1 Means For Simulation Runs

CM = 187171.6   SST = 458.33

ANOVA

TREATMENT	DF	SS	MS	F
COLUMN	2	244.5	122.25	13.74
ROW	2	178.23	89.12	10.0
ERROR	4	35.6	8.9	
TOTAL	8	458.33		

$F_{2,4,.05}=6.94$

Figure D-2 Two Way Anova Of Means

Vita

Second Lieutenant James E. Bennett was born in Greenville, Mississippi on 29 April 1958. He received a Bachelor of Arts degree in Mathematics from the University of Mississippi in May 1980. He received his commission in the United States Air Force at that time. Immediately following graduation, Lieutenant Bennett reported to the School of Engineering at the Air Force Institute of Technology. While there he received a Masters of Science in Operations Research, graduating in December 1981. Lieutenant Bennett is married to the former Cindy Joann Miley and they have one daughter, Allison Martha.

Permanent Address: 142 Waywood Drive
Jackson, Mississippi 39212

END

DATE
FILMED

3-82

DTIC